A Time Series Prediction Using LSTM Networks for Prediction of Covid-19 Data

Mrs. K. Sreelatha[1], D.Srihitha[2], SK. Sameer[3], V. Nikhila[4]
[1,2,3,4]Dept of ECM, Sreenidhi Institute of Science and Technology, Hyderabad, India

Abstract - **The new episode of Coronavirus sickness 2019 (COVID-19), which gets brought about by extreme intense respiratory condition (SARS) Covid 2 (SARS-CoV-2), has been answerable for the passing of more than 3,00,000 individuals and simultaneously has contaminated over 4.7 million individuals in the entire world as of mid-May 2020. There has been more that 1.8 million recuperations during this period as well. It gets basic for Governments to know about the circumstance and to have the option to foresee the future number of patients so availability as far as medical care and arranging of other fundamental activities can be kept up. Utilizing this as a main impetus, a model for expectation of the quantity of COVID–19 patients has been created utilizing the Long-Sort Term Memory (LSTM) organization and afterward utilized it for estimating future cases. The cases India is considered. The investigation tracks down that the LSTM network created in this paper performs better compared to different organizations and subsequently can be a helpful contender for expectation of future number of patients of COVID–19.**

Index Terms - **Machine learning algorithms, Coronavirus, long short-term memory networks, training and testing data.**

I.INTRODUCTION

In the current situation there has been no treatment strategy or immunization present, and the solitary shields against this pandemic are social removing and great hand cleanliness. Numerous nations have utilized lock-down to force the social separating and reducing local area spread that has hindered the spread. Be that as it may, this activity has brought about a major monetary log jam and can't be the drawn-out answer for this pandemic. Right now, it is a significant wellbeing emergency all throughout the planet and it would not be right to say that it is 'a foe to mankind'. In the present situation, the solitary alternative is forestalling the event of contamination and setting up our medical services framework for the likely up-comings. The essential thought process in the undertaking is to encourage the framework which is valuable for Coronavirus cases forecast with improved exactness than the current techniques. It includes anticipating the complete number of Coronavirus cases in future with high precision rate from the diverse characterization calculations executed. The information pre- preparing is performed on the dataset which would include measure like information cleaning, information scaling and information marking. This venture is carried out utilizing the Indian Covid cases dataset. This is the dataset taken from the WHO site. With the accessible standard dataset, fitting information pre-preparing procedures and executing Long Short-Term Memory neural organizations It helps in the forecast future Coronavirus cases which makes the conclusion cycle simpler and faster. The fundamental goal of the task is to predict the Coronavirus cases utilizing LSTM model. Because of the exactness of the model, it turns out to be more effective and reliable. We foster a code which helps in the assurance of chart of affirmed cases and furthermore precision which demonstrates that it is superior to other old-style calculations. This framework can be utilized directly by the end client.

II. RELATED WORK

From this paper, we represent some of the related works for the prediction of deadly diseases like pneumonia, COVID-19 and other chest-oriented diseases. They have taken a dataset comprising Xray images from victims suffering with breathing disorders, COVID-19, and other diseases.[1] They used Google-Net of CNN, for the detection of irregularity in medical X-ray images shows outstanding results upto 96%. They concluded that Deep Learning from X-ray images can successfully trace significant distinct biomarkers associated with Coronavirus(SARS-Co-2). They used an unsupervised classification algorithm namely fuzzy c-means for identification of Pneumonia disease in chest X-ray images . Unsupervised fuzzy C-means approach was observed to give better identification results than the rest of the ways like DWT, WFT, and WPT.[1] They proposed to calculate the ratio of area of healthy respiratory organ region to total respiratory organ region to identify the presence of Pneumonia and Covid-19. They also used Convolutional Neural Networks for Pneumonia detection from X-ray images and showed the classification accuracy as 84%.

In this paper, the authors performs both training and testing according to the time using AI methods of ANN, training the data with Grey Wolf Optimizer(GWO). In this the authors, clearly explained about both the algorithm used ANN and optimizer.[2] As we know ANN mainly deals with non-linear problems and predict the outputs based on external world(input signals).whereas GWO is a recent swarm intelligence algorithm as it is an optimization technology. We also noticed the result analysis validated and evaluated based on mean absolute error (MAPE Factor).[2] The time period for the both testing and training of data is as follows i.e; (Jan 22 - Oct 15,2020).According to outputs the selected ANN architecture with a 150 at a most iteration of 500 provided a well accuracy for all the phases.

With this paper, we came to know that regression models also play crucial role in analyzing and predicting the coronavirus(SARS-Co-2).[3]Even Time series forecasting od different cases also used in such a way which includes data collection, data processing, data visualization, implementation of the models and their results. In this paper, they used Linear regression and polynomial regression and the result analysis are depends on the R^2 score and mean absolute error. [3] And finally they concluded that polynomial regression serves better than that of linear regression. However, they also said that we can decrease the error rate in future times by taking bigger data sets, using better algorithms and fine tuning of the attributes.

The main objective of this paper is to predict the usage of health services and mortality rate of every part in the USA over the next three months.[4] The dataset was taken from local and national government websites and the WHO. The modeling and approach was a four-step process: (i) pointing out and filtering of COVID-19 data; (ii) statistical model approximation for mortality rates as a function of time since the rate exceeds a threshold in a place; (iii) predicting time to exceed a given death threshold in places in the beginning of pandemic; and (iv) modeling health services usage as a reason of deaths.[4] In the end, the results predicted a rise in demand for the health services in the last weeks of March and April and slowly decreasing with a constant demand until June. The author used LSTM, CNN, Decision tree algorithms in solving this time series regression problems.

In this particular paper, we observe that they used several models to forecast the vast spreading of disease. But among all those epidemic models the most commonly used is Susceptible-Infectious-Recovery(SIR)epidemic model. It creates different variants of SIRD model to acquire the best outcomes of forecast.[5] After trying all the variants, it is observed that the combination of 2 particular variants is giving the best values. The model was fitted with available data of disease till 11 May 2020 to determine the value of attributes δ, β and γ. When the values were determined, a graph was plotted to forecast the future cases of disease(COVID 19) spreading among people.[5] Depending on the model, it can be seen that the number of suscepts being infected from COVID will reduce and the situation will improve by the month of August and September. They also said the cases may reach exponentially to the peak value. But finally, they concluded that the predicted and the actual cases that reported based on data till 20 May 2020,were not same. So they suggest for further forecasting based on SIRD.

III.PROPOSED SOLUTION

The proposed system architecture is concerned with the input data, the trained model, and the output. The user provides feature values as input and these values are processed by the trained model. In addition, the output is presented to the user in the form of a graph. The data set is divided into training dataset and validation dataset. Two datasets are used to generate the appropriate classifier for model training and further performance evaluation. Therefore, we get results in the trained model. We can provide the input and then predict the output. An architecture diagram is a diagram of a system used to summarize the general outline of a software system and the relationships, constraints, and boundaries between components. It is an important tool because it gives an overview of the physical implementation of the software system and its development path.

We implemented a simple long-term short-term memory (LSTM) model with an input layer, a single hidden layer, and an output layer used to make predictions.

Data preprocessing and feature engineering should be done before building the LSTM model.

- Create a dataset, make sure all the data is float.
- Standardize characteristics.
- Concatenate into training and test sets.
- Convert an array of values to a data set matrix.
- Model in X = t and Y = t + 1.
- Model entry into 3D (num_samples, num_timesteps, num_features).

The input layer has neurons by 5 sequence steps. The hidden layer is an LSTM layer with 10 hidden units (neurons) and one rectified linear unit (ReLU) as an activation function. The output layer has a dense layer with 1 unit to predict the output. The test rate is set to 0.2 and it decreases every five centuries. Also, we used 300 as the number of epochs, Adam as the optimizer and square root error as the loss function. Then we fit the model with the prepared data to make predictions. The results obtained may vary due to the random nature of the LSTM model; so we've done it many times. Finally, we enter the last sequence with the output to predict the next value in the sequence.

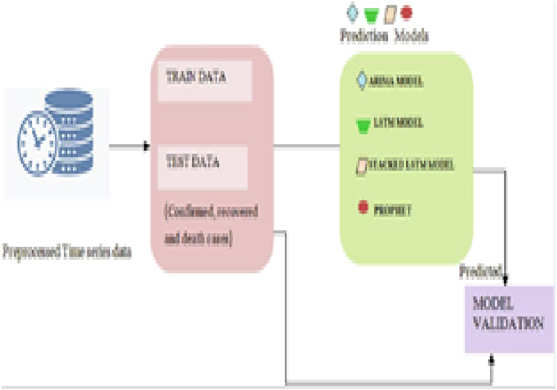


Fig 3.1 Proposed Model

IV METHODOLOGY

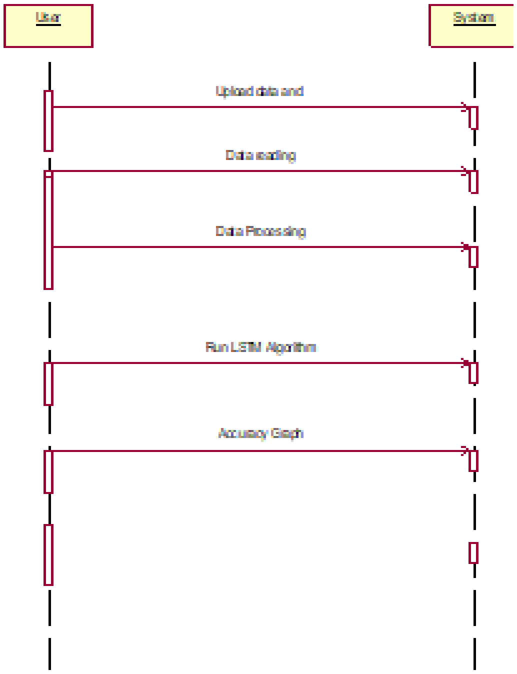
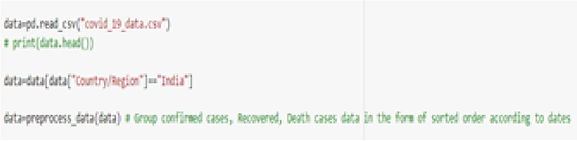


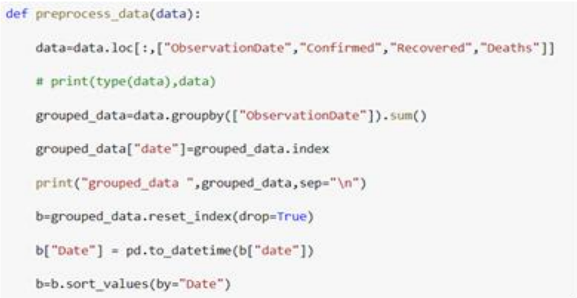
Fig 4.1 Flowchart

Stage 1: Data Preprocessing and grouping

Data pre-processing is the most crucial component of any machine learning system. There may be several types of data that must be homogenised in accordance with the technologies employed. In the instance of our model, the data pre-processing removes missing values, duplicate values, and null values. Outliers are also deleted, and we choose data from a certain region based on our needs. Because our collection is called World Covid, it contains covid case records from all around the world. However, because we are only interested in Indian covid instances in our project, we must extract Indian data during preprocessing. We must also group our data by observation dates because it is timeseries data.



```
data=pd.read_csv("covid_19_data.csv")
# print(data.head())

data=data[data["Country/Region"]=="India"]

data=preprocess_data(data) # Group confirmed cases, Recovered, Death cases data in the form of sorted order according to dates
```



```
def preprocess_data(data):

    data=data.loc[:,["ObservationDate","Confirmed","Recovered","Deaths"]]

    # print(type(data),data)

    grouped_data=data.groupby(["ObservationDate"]).sum()

    grouped_data["date"]=grouped_data.index

    print("grouped_data ",grouped_data,sep="\n")

    b=grouped_data.reset_index(drop=True)

    b["Date"] = pd.to_datetime(b["date"])

    b=b.sort_values(by="Date")
```

We turn each column into a list, such as date list for observation dates, confirmed list for confirmed covid instances, and so on, after grouping the data by observation dates.

Stage 2:Data Preparing

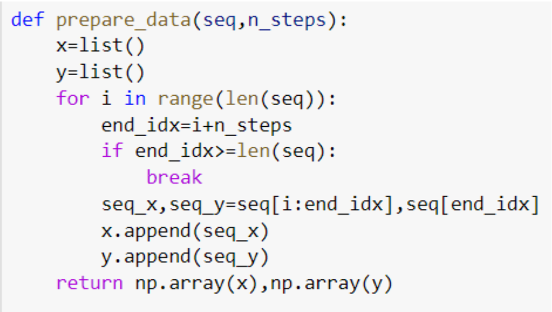
A LSTM series must first be prepared before it can be modelled. This algorithm trains the model that transfers the previous scenarios as input to a new observation as output. As a result, the series of inputs should be converted.

Take the following uni-variate sequence as an example:

[11, 12, 13, 14, 15, 16, 17, 18, 19]

For the one-step prediction that is being learnt, we have divided the series into two formats as input and output with differentiating intervals of time. The outputs are given as inputs for future cases.

P	Q
[11, 12, 13]	14
[12, 13, 14]	15
[13, 14, 15]	16
...	

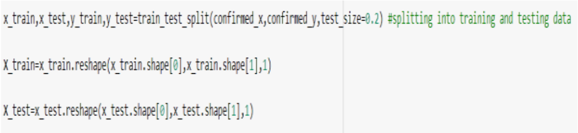


```
def prepare_data(seq,n_steps):
    x=list()
    y=list()
    for i in range(len(seq)):
        end_idx=i+n_steps
        if end_idx>=len(seq):
            break
        seq_x,seq_y=seq[i:end_idx],seq[end_idx]
        x.append(seq_x)
        y.append(seq_y)
    return np.array(x),np.array(y)
```

Fig 4.3 Data Preparing

Stage 3: Splitting the Data

The important aspect of the prediction models is that before loading the model, we must first divide our data into huge portions of training and a partial testing portions. The data considered for training is then used to explain/analyse the model’s behaviour, and the testing data is for examining the model. During the model's testing, we receive the expected and actual outputs, which will be utilized for calculating the accurate behaviour later. We have split major portion (80%) of the input as training set and minor portion (20%) by setting split size to 0.2 for testing in our project.

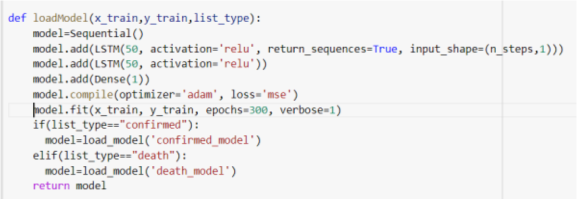


```
x_train,x_test,y_train,y_test=train_test_split(confirmed_x,confirmed_y,test_size=0.2) #splitting into training and testing data

X_train=x_train.reshape(x_train.shape[0],x_train.shape[1],1)

X_test=x_test.reshape(x_test.shape[0],x_test.shape[1],1)
```

Fig 4.4 Splitting of Dataset

Step 4: Loading Model (LSTM)

Long Short-Term Memory (LSTM) model comprising an input layer, one hidden layer, and an output layer that makes a prediction has been built. Some neurons are present in the input layer are equal to 5 sequence steps. It has various layers for instance 10 hidden memory layers and a ReLU layer for activation over the above mentioned layer. For anticipating the output, the output layer has only one unit of layer. The testing rate here considered is 0.2, with a five-epoch decay. Furthermore, epochs can be raised to 300, Adam was the optimizer, and the loss function was the mean square error. The model was then fitted to the data in order to make a forecast. Because of the stochastic nature of the LSTM model, the generated results may vary; as a result, we ran it numerous times. Finally, we have inputted the output from the past sequence to predict the next value in the series.



```
def loadModel(x_train,y_train,list_type):
    model=Sequential()
    model.add(LSTM(50, activation='relu', return_sequences=True, input_shape=(n_steps,1)))
    model.add(LSTM(50, activation='relu'))
    model.add(Dense(1))
    model.compile(optimizer='adam', loss='mse')
    model.fit(x_train, y_train, epochs=300, verbose=1)
    if(list_type=="confirmed"):
      model=load_model('confirmed_model')
    elif(list_type=="death"):
      model=load_model('death_model')
    return model
```

Fig 4.5 Long Short term memory networks algorithm

Step 5: After that, the model gets ready for prediction of confirmed and death cases, accuracy of the model is also obtained with the output graphs in the final stage.

V RESULTS AND DISCUSSION

In our study, we attempted to forecast future covid cases and death cases using an LSTM algorithm with a higher accuracy than the present system. Data pre-processing and data grouping aid in the preparation of the dataset and the acquisition of superior insights. We determine the model's accuracy as well as the prediction of future covid situations after putting the dataset on the machine learning model.

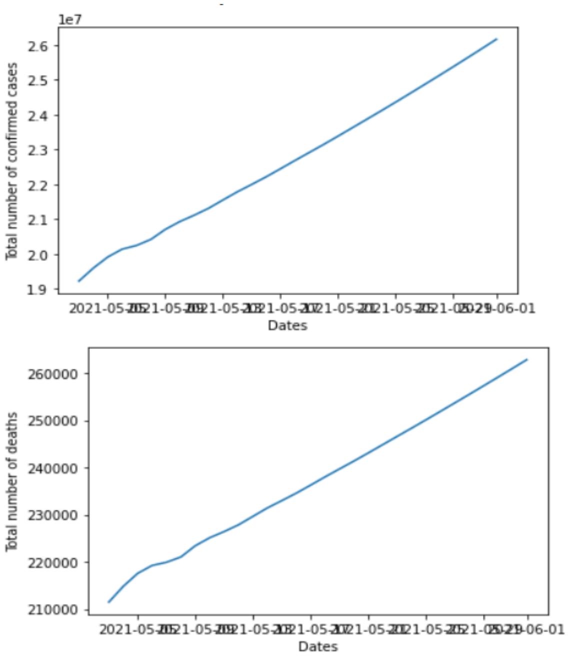
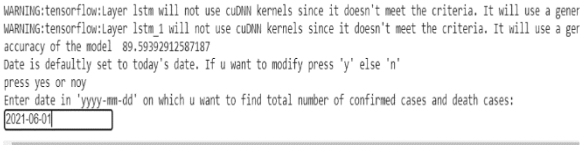


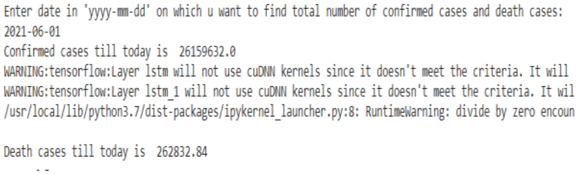
Fig 5.1 : Results

The Visualization methods in order to show the output graph for confirmed and death cases.



```
WARNING:tensorflow:Layer lstm will not use cuDNN kernels since it doesn't meet the criteria. It will use a gener
WARNING:tensorflow:Layer lstm_1 will not use cuDNN kernels since it doesn't meet the criteria. It will use a ger
accuracy of the model  89.59392912587187
Date is defaultly set to today's date. If u want to modify press 'y' else 'n'
press yes or noy
Enter date in 'yyyy-mm-dd' on which u want to find total number of confirmed cases and death cases:
2021-06-01
```

Fig5.1.1: Results of observed Accuracy of the model used in our project.



```
Enter date in 'yyyy-mm-dd' on which u want to find total number of confirmed cases and death cases:
2021-06-01
Confirmed cases till today is  26159632.0
WARNING:tensorflow:Layer lstm will not use cuDNN kernels since it doesn't meet the criteria. It will
WARNING:tensorflow:Layer lstm_1 will not use cuDNN kernels since it doesn't meet the criteria. It wil
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:8: RuntimeWarning: divide by zero encoun

Death cases till today is  262832.84
```

Fig 5.2 Results of expected confirmed and death cases for a particular date given by the user.

VI.CONCLUSION AND FUTURE SCOPE

The forecasting capability of the COVID-19 dataset was investigated using an LSTM network in this study. By providing a high accuracy rate, the LSTM network outperformed the other competing networks. Hence, we can say our model is outperforming other sources in terms of accuracy. In this model, we have achieved over 80-90% of accuracy both for death and future confirmed cases.

Coronavirus, like the flu, is a contagious and infectious disease with specific growth patterns. These patterns are non-linear and dynamic in character. The data is dynamic in nature because the cases may vary depending on the seasons, population, and other factors. For better performance and reliable results, we can use this model to predict the count.This model uses memory networks in terms of time series, that helps prediction in getting realistic and closer to the actual scenario. This could alleviate stress on health-care systems and administrations by allowing them to plan more effectively. As a result, the LSTM model could be a good contender for predicting the number of COVID–19 patients in the future.

This work can be further enhanced in the future by further development of algorithms that are trained on the highly variety of data allowing the algorithm to predict with greater accuracy than that is achieved now. This way the aim of predicting right number can become more reliable and thereby helping in taking precautionary measures in earlier stages and increasing the chances of curing the highly successful.

VII.ACKNOWLEDGMENT

This work was supported by the Department of Electronics and Computer Engineering, Sreenidhi Institute of Science and Technology, we thank the Head of Department, Dr. D. Mohan, for his guidance and support.

REFERENCES

[1] D. Haritha, N. Swaroop and M. Mounika, "Prediction of COVID-19 Cases Using CNN with X-rays," 2020 5th International Conference on Computing, Communication and Security (ICCCS), 2020, pp. 1-6, doi: 10.1109/ICCCS49678.2020.9276753.

[2] S. Ardabili, A. Mosavi, S. S. Band and A. R. Varkonyi-Koczy, "Coronavirus Disease (COVID-19) Global Prediction Using Hybrid Artificial Intelligence Method of ANN Trained with Grey Wolf Optimizer," 2020 IEEE 3rd International Conference and Workshop in Óbuda on Electrical and Power Engineering (CANDO-EPE), 2020, pp. 000251-000254, doi: 10.1109/CANDO-EPE51100.2020.9337757.

[3] S. Shaikh, J. Gala, A. Jain, S. Advani, S. Jaidhara and M. Roja Edinburgh, "Analysis and Prediction of COVID-19 using Regression Models and Time Series Forecasting," 2021 11th International Conference on Cloud Computing, Data Science &

Engineering (Confluence), 2021, pp. 989-995, doi: 10.1109/Confluence51648.2021.9377137.

[4] A. Kunjir, D. Joshi, R. Chadha, T. Wadiwala and V. Trikha, "A Comparative Study of Predictive Machine Learning Algorithms for COVID-19 Trends and Analysis," 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2020, pp. 3407-3412, doi: 10.1109/SMC42975.2020.9282953.

[5] S. Singh, P. Raj, R. Kumar and R. Chaujar, "Prediction and forecast for COVID-19 Outbreak in India based on Enhanced Epidemiological Models," 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), 2020, pp. 93-97, doi: 10.1109/ICIRCA48905.2020.9183126.

[6] World Health Organization, Coronavirus disease (COVID-19) outbreak. https://www.who.int/emergencies/diseases/novel-coronavirus-2019 (accessed on May 14, 2020).

[7] Zhu, N.; Zhang, D.; Wang, W.; Li, X.; Yang, B.; Song, J.; Zhao, X.; Huang, B.; Shi, W.; Lu, R.; Niu, P.; Zhan, F.; Ma, X.; Wang, D.; Xu, W.; Wu, G.; Gao, G. F.; Tan, W. A novel coronavirus from patients with pneumonia in China, 2019. N. Engl. J. Med. 2020, 382, 727.

[8] Paules, C. I.; Marston, H. D.; Fauci, A. S. Coronavirus infections—more than just the common cold, JAMA 2020, 323, 707.

[9] Johns Hopkins University Center for Systems Science and Engineering, Coronavirus (COVID-19) Cases. https://github.com/CSSEGISand Data/COVIDQ3 (accessed on April 14, 2020).

[10] https://www.who.int/emergencies/diseases/novel-coronavirus-2019/global-research-on-novel-coronavirus-2019-ncov. (Accessed on May 14, 2020)

[11] N. C. P. E. R. E. Team. The epidemiological characteristics of an outbreak of 2019 novel coronavirus diseases (COVID-19) in China, China CDC Weekly 2020, 41, 145.

[12] W. Jiang and H. D. Schotten, "Deep Learning for Fading Channel Prediction," in IEEE Open Journal of the Communications Society, vol. 1, pp. 320-332, 2020